

Generate BDD Tests (in Gherkin format) for the given URL

Problem Statement

You are required to design and implement an AI-based automation solution that dynamically generates Gherkin-style testing scenarios for websites containing hovering elements (for example: navigation menus, image overlays, tooltip popups, or dropdown lists revealed on hover).

Objective

The system should :

1. Accept a website URL as input and
2. Automatically:
 - a. Detect and analyze hoverable elements across the given page (not limited to menus).
 - b. Simulate hover actions using browser automation. (e.g., Playwright).
 - c. Identify elements or links that become visible as a result of the hover interaction.
 - d. Identify any popup/modal behaviors triggered by actions.

Generate clean, well-formatted Gherkin scenarios describing how a tester should validate:

- The visibility and behavior of hover-based elements.
- The clickability and redirection (if applicable) of revealed links or elements.

⚠ Important Note:

The solution must be **fully dynamic and autonomous**.

It should **not rely on any hardcoded selectors, labels, or element identifiers**.

Instead, it should **analyze the webpage structure and behaviors** to understand interactions at runtime.

It should work seamlessly across different types of modern, dynamic websites — not just static pages or predefined DOM structures.

Functional Expectations / Requirements

- Input: A single website URL.
- Output: Automatically generated **Gherkin .feature file** (or console output) containing two dynamic test cases for that URL:
 - a. Validation of popup/overlay behavior (if found).
 - b. Validation of hover-based interactions (e.g., dropdowns or hover menus).

Tech Stack Requirements

Development must be done in Python. Refer the following suggested technologies. But you are free to use any other technical stack as well, as long as it is Python based solution.

Suggested Technical Stack

Layer	Recommended Tools
Browser Automation	Playwright / Selenium
HTML Parsing / DOM Analysis	BeautifulSoup, lxml
LLM Reasoning	OpenAI GPT-4/GPT-5, Gemini, or Ollama (local)
Agentic Frameworks	Autogen / CrewAI / LangGraph / MCP
Backend API	FastAPI
UI (Optional)	Streamlit
Storage	SQLite or JSON for caching DOM metadata
Output Format	.feature files (Gherkin)

Candidate may use one or multiple of the above frameworks as part of the solution.

Use of open-source libraries or projects is allowed if appropriately cited or integrated.

Example Website

Input URL's:

1. <https://www.tivdak.com/patient-stories/>
2. <https://www.nike.com/in/>
3. <https://www.apple.com/in/>

Expected Example Gherkin Scenarios

Normal Test Steps	Gherkin BDD Format
<p>Test 1:</p> <ol style="list-style-type: none"> 1. Goto the url https://www.tivdak.com/patient-stories/ 2. Click the button called "Learn More" 3. A pop up will appear with the title saying "You are now leaving tivdak.com" 4. Click the cancel button 5. Click the button called "Learn More" 6. A pop up will appear with the title saying "You are now leaving tivdak.com" 7. Click the continue button 8. Verify the url has changed to "https://alishasjourney.com/" 	<p>Feature: Validate "Learn More" pop-up functionality</p> <p>Scenario: Verify the cancel button in the "You are now leaving tivdak.com" pop-up</p> <p>Given the user is on the "https://www.tivdak.com/patient-stories/" page</p> <p>When the user clicks the "Learn More" button</p> <p>Then a pop-up should appear with the title "You are now leaving tivdak.com"</p> <p>And the user clicks the "Cancel" button</p> <p>Then the pop-up should close and the user should remain on the same page</p> <p>When the user clicks the "Learn More" button</p> <p>Then a pop-up should appear with the title "You are now leaving tivdak.com"</p> <p>And the user clicks the "Continue" button</p> <p>the same page</p> <p>Then the page URL should change to "https://alishasjourney.com/"</p>
<p>Test 2:</p> <ol style="list-style-type: none"> 1. Go to the URL https://www.tivdak.com/patient-stories/ 	<p>Feature: Validate navigation menu functionality</p>

Normal Test Steps	Gherkin BDD Format
<p>2. Hover over the navigation menu "About Tivdak"</p> <p>3. A drop-down will appear</p> <p>4. Click on the link "What is Tivdak?"</p> <p>5. Verify the URL has changed to "https://www.tivdak.com/about-tivdak/"</p>	<p>Scenario: Verify navigation from "Patient Stories" to "What is Tivdak" page</p> <p>Given the user is on the "https://www.tivdak.com/patient-stories/" page</p> <p>When the user hovers over the navigation menu "About Tivdak"</p> <p>And clicks the link "What is Tivdak?" from the dropdown</p> <p>Then the page URL should change to "https://www.tivdak.com/about-tivdak/"</p>

Deliverables:

- Source code (well-structured and modular)
- Push the complete solution to a public GitHub repository and share the link
- Include a well-structured README file with setup, usage instructions, and requirements.txt
- Provide a demo video (max. 7 minutes) showcasing the solution flow and output and key code logic.
- Sample output Gherkin file (e.g., hover_tests.feature) or console output
- The solution should cover both logics — detecting hover elements and popup elements.

References

BDD Gherkin - <https://cucumber.io/docs/gherkin/>

Please refer the above table. The same tests written in Gherkin format.